DOI: 10.36910/6775-2524-0560-2020-40-19
УДК: 004.05(075.8)
Марценюк Василь Петрович[1], д.т.н., професор,
https://orcid.org/0000-0001-5622-1038
Сверстюк Андрій Степанович[2], к.т.н., доцент,
https://orcid.org/0000-0001-8644-0776
Андрущак Ігор Євгенович[3], д.т.н., професор,
https://orcid.org/0000-0002-8751-4420
Сіваковська Олена Миколаївна[3], к.т.н., доцент.
https://orcid.org/0000-0002-9300-0039
Потейчук Михайло Іванович[3], аспірант.
https://orcid.org/0000-0001-7263-0958
Університет Бєльсько-Бяли, Польща[1)]
Тернопільський національнийний медичний університет імені І.Я. Горбачевського, Україна[2)]
Луцький національний технічний університет, Україна[3)]

FEATURES OF MULTIFUNCTIONAL BACKDOOR TECHNOLOGY IN THE PERSONAL SPACE OF USERS

Марценюк В.П., Сверстюк А.С., Андрущак І.Є., Сіваковська О.М., Потейчук М.І. Особливості багатофункціональної технології Backdoor в особистому просторі користувачів. У статті розглядається проблема бекдору, як методу обходу стандартних процедур аутентифікації, несанкціонованого віддаленного доступу до комп'ютера, отримання доступу до відкритого тексту, і так далі, залишаючись при цьому непоміченим. Проаналізовано сучасні форми та функції вторгнення бекдору.
Ключові слова: бекдор, аутентефикація, несанкціонованій доступ, криптосистема.

Марценюк В.П., Сверстюк А.С., Андрущак И.Е., Сиваковська Е.Н., Потейчук М.И. Особенности многофункциональной технологии Backdoor в личном пространстве пользователей. В статье рассматривается проблема бэкдора, как метода обхода стандартных процедур аутентификации, несанкционированного удаленного доступа к компьютеру, доступа к открытого текста, и так далее, оставаясь при этом незамеченным. Проанализированы современные формы и функции вторжения бэкдора.
Ключевые слова: бэкдор, аутентефикация, несанкционований доступ, криптосистема.

Martsenyuk V.P., Sverstiuk A.S., Andrushchak I.Ye., Sivakovska O.M., Poteichuk M.I. Features of multifunctional Backdoor technology in the personal space of users. The article considers the problem of backdoor as a method of bypassing standard authentication procedures, unauthorized remote access to a computer, gaining access to plaintext, and so on, while remaining unnoticed. Modern forms and functions of backdoor invasion are analyzed.
Keywords: backdoor, authentication, unauthorized access, cryptosystem.

Formulation of the problem. A backdoor is malware that is used by cybercriminals to gain unauthorized remote access to a computer system through a security vulnerability. The backdoor runs in the background and hides from the user. It is very similar to other malicious viruses and is therefore quite difficult to detect. A backdoor is one of the most dangerous types of parasites, as it gives hackers the ability to perform any possible action on an infected computer. An attacker can use a backdoor to spy on a user, manage their files, install additional software or dangerous threats, control the entire PC system, and attack other hosts. The backdoor often has additional, destructive capabilities, such as taking screenshots, infecting and encrypting files. Such a parasite is a combination of various, secret and secure threats that work on their own and do not require control at all.

Analysis of research. We live in a society where information and access to it are very important. Personal passwords, e-mail addresses, any personal information becomes the object of increased attention and value. Many companies are interested in "live" email addresses to which you can send advertising, information about the sites you visit the most, and so on. It is clear that obtaining this kind of information can be done, for example, in the framework of sociological research. However, sometimes to get it go through the use of covert cyber espionage programs. Malicious programs are created that spy on users and send the received personal information to certain sites, or secretly open access to personal PCs to interested persons. This is a kind of spyware that belongs to such types as backdoor and spyware.

The main purpose of Backdoor is covert computer control. Backdoors can take completely different forms. For example, this may be part of a regular program that additionally has a share of malicious code. By the way, the function of saving passwords "by default" can act as a kind of Backdoor, if, of course, the user has not changed them yet.. Problems with the use of Backdoor were raised at AFIPS conferences to discuss computer sabotage. The scale of the threat grew, and this aspect of computer security was addressed separately in a RAND Corporation

report. In particular, according to media reports, Backdoor was found in a number of Samsung products on the Android system. The OS of some models of phones of this brand contained a backdoor, which opened remote access to data stored on the device itself. As for Spyware, it is used to transfer information from your PC to third parties, and you may not even know it. Such information base can also be used for selfish purposes, for example, to be sold to companies specializing in spam and advertising materials.

Presentation of the main material and the justification of the results. Most backdoors are malware that must somehow infiltrate the computer. However, some parasites do not require installation, as parts of them are already integrated into the software that runs on the remote host. Programmers sometimes leave such backdoors in their software for diagnostics and troubleshooting purposes. But in reality, hackers only use them to hack into the system. Generally speaking, backdoors are specific Trojans, viruses, keyloggers, spyware, and remote administration tools. They work in the same way as the mentioned virus applications do. However, their functions and loads are more complex and dangerous, therefore, they are grouped into one special category.

There are two types of backdoors currently most common. The first is client-server backdoors. In such a backdoor, it is possible, generally speaking, to distinguish two whole programs - the first of them is secretly installed on the affected computer, and the second, as you probably already guessed, is used to remotely control the first of them and is installed, respectively, on the attacker's computer. Backdoors of the second type use a built-in client for remote control, working via Telnet, HTTP, or IRC. To manage such a backdoor, of course, no special client software is required. Currently, cybercriminals are actively using both types of backdoors.

There are a number of hidden considerations to keep in mind when assigning responsibilities.Latent backdoors are sometimes disguised as unintentional defects (errors) for plausible denial reasons. In some cases, they may start out as a real mistake (unintentional error), which, once discovered, then deliberately remains unrecorded and not disclosed, whether it is a fraudster for reasons of personal gain or due to the knowledge and oversight of senior managers.

It is also possible that the technology base of a fully open corporation could be secretly and reliably tainted by outside agents (hackers), although this level of complexity is believed to exist primarily at the level of national state actors. For example, if a photomask received from a photomask vendor differs by several points from its photomask specification, it will be difficult for the chip maker to detect this if it is otherwise functionally silent; A hidden rootkit running in photomask etching equipment can also cause this mismatch without the knowledge of the photomask manufacturer, and thus one backdoor potentially leads to another. (This hypothetical scenario is essentially a silicon version of the undetectable compiler backdoor discussed below.) In general, the long chains of dependencies in today's highly specialized technology economy and the myriad of human-based process control points make it difficult to definitively define responsibility at a time when a hidden backdoor is discovered. Even an outright admission of responsibility must be scrutinized if the admitted party owes other influential interests. Backdoors are unable to spread and infect a system without the user's knowledge. Most of these parasites need to be installed manually in conjunction with other software.

There are four main ways these threats enter the system:

- Unknowing PC users can accidentally install typical backdoors on their computers. They can come attached to email messages or file sharing programs. The authors give them unsuspecting names and trick the user to open or run such a file.

- Backdoors are often installed by other parasites such as viruses, Trojans or even spyware. They enter the system without the knowledge and permission of the user who uses the infected computer. Some threats can be installed manually by hackers who have sufficient privileges to install software. A small proportion of backdoors can be spread through the use of remote systems with some security vulnerabilities.

- Several backdoors are already integrated into specific applications. Even legitimate programs can be tampered with by remote access functions. The attacking file must communicate with the computer through the installation of such a program in order to instantly gain access to the system or take control of certain software.

- Some backdoors infect computers by exploiting certain software vulnerabilities. They work just like worms and automatically spread without the user's knowledge. The user cannot notice anything suspicious as the threats do not display any setup wizards, dialog boxes or warnings.

The widespread distribution of backdoors mainly infects computers running the Microsoft Windows operating system. However, many of the less common parasites are designed to work in various fields, such as the Mac operating system [1].

The backdoor allows hackers to treat the infected computer as if they were their own PC and use it for various malicious purposes or even criminal activity. In most cases, it is really difficult to figure out who is controlling the parasite. In fact, backdoors are very difficult to detect. They can violate user privacy for months

or even years until the user notices them. An attacker can use a loophole to find out everything about a user, obtain and reveal invaluable information such as passwords, logins, credit card numbers, exact bank account details, valuable personal documents, contacts, interests, web browsing habits, and more. Backdoors can be used for destructive purposes. If a hacker was unable to obtain some valuable and useful information from the infected computer, or has already stolen it, in the end, he can destroy the entire system in order to destroy his tracks. This means that all hard drives will be formatted and all files on them will be permanently deleted.

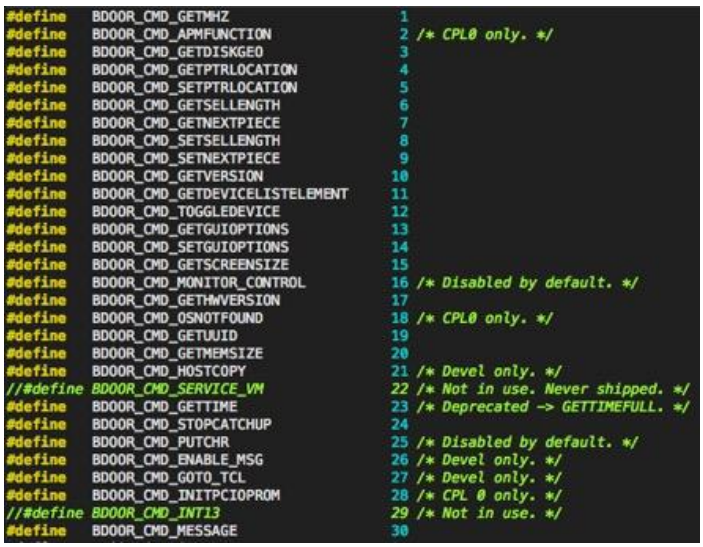
When the backdoor finds the path to the system, it calls the following actions:

- allows an attacker to create, delete, rename, copy or edit any file, execute various commands, change any system settings, modify the Windows registry, launch, monitor and eliminate applications, install other software;
- allows a hacker to control the hardware devices of the computer, change the settings related to shutdown or restart of the computer without permission;
- steals personal information, valuable documents, passwords, logins, identity data, user activity logs and monitors web browsing habits;
- records button presses and takes screenshots. In addition, it sends the collected data to certain email addresses, uploads it to a specified FTP server or transmits it via an Internet connection to remote hosts;
- infects files, installed applications and damages the entire system;
- distributes infected files to remote computers with some security vulnerabilities, performs attacks against hackers on remote hosts;
- installs a hidden FTP server that can be used by malefactors for various illegal purposes [2].

There are many different backdoors. The following examples illustrate how functional and extremely dangerous these parasites can be.

FinSpy is a backdoor that allows a remote attacker to download and run any file from the Internet. The parasite reduces the overall security of the system by changing the default settings of the Windows firewall and initiates other system changes. FinSpy relies on files that use random names, so it is quite difficult to find its loophole and remove it from the system. The backdoor starts automatically every time Windows starts, and can only be stopped with updated anti-spyware software. Tixanbot is an extremely dangerous backdoor that gives a hacker full access to an infected computer. An attacker can control the entire system and files, download and install arbitrary applications, update the backdoor, change Internet Explorer home page settings, attack remote hosts, and obtain system information. Tixanbot shuts down and processes essential system services and security software, closes active spyware washes, and removes registry entries associated with firewalls, antivirus and antispyware software to prevent them from starting at Windows startup. The parasite also blocks access to reputable security-related web resources. Tixanbot is redistributable, it sends messages with specific links to all MSN contacts. By clicking on such a download link, the backdoor is installed (Pic 1).

Briba is a backdoor that gives a hacker remote and unauthorized access to an infected computer system. This parasite launches a hidden FTP server that can be used to download, update, or run malicious software. Briba's actions can lead to visible instability, computer malfunction and privacy breaches. [3]



```
#define   BDOOR_CMD_GETMHZ                1
#define   BDOOR_CMD_APMFUNCTION           2 /* CPL0 only. */
#define   BDOOR_CMD_GETDISKGEO            3
#define   BDOOR_CMD_GETPTRLOCATION        4
#define   BDOOR_CMD_SETPTRLOCATION        5
#define   BDOOR_CMD_GETSELLENGTH          6
#define   BDOOR_CMD_GETNEXTPIECE          7
#define   BDOOR_CMD_SETSELLENGTH          8
#define   BDOOR_CMD_SETNEXTPIECE          9
#define   BDOOR_CMD_GETVERSION            10
#define   BDOOR_CMD_GETDEVICELISTELEMENT  11
#define   BDOOR_CMD_TOGGLEDEVICE          12
#define   BDOOR_CMD_GETGUIOPTIONS         13
#define   BDOOR_CMD_SETGUIOPTIONS         14
#define   BDOOR_CMD_GETSCREENSIZE         15
#define   BDOOR_CMD_MONITOR_CONTROL       16 /* Disabled by default. */
#define   BDOOR_CMD_GETHWVERSION          17
#define   BDOOR_CMD_OSNOTFOUND            18 /* CPL0 only. */
#define   BDOOR_CMD_GETUUID               19
#define   BDOOR_CMD_GETMEMSIZE            20
#define   BDOOR_CMD_HOSTCOPY              21 /* Devel only. */
//#define BDOOR_CMD_SERVICE_VM            22 /* Not in use. Never shipped. */
#define   BDOOR_CMD_GETTIME               23 /* Deprecated -> GETTIMEFULL. */
#define   BDOOR_CMD_STOPCATCHUP           24
#define   BDOOR_CMD_PUTCHR                25 /* Disabled by default. */
#define   BDOOR_CMD_ENABLE_MSG            26 /* Devel only. */
#define   BDOOR_CMD_GOTO_TCL              27 /* Devel only. */
#define   BDOOR_CMD_INITPCIOPROM          28 /* CPL 0 only. */
//#define BDOOR_CMD_INT13                 29 /* Not in use. */
#define   BDOOR_CMD_MESSAGE               30
```

Pic 1. Backdoor test

A site backdoor is a small, well-disguised code in existing site files that allows an attacker to gain full access to a server by downloading web shells. For more information about backdoors on the site, see the page about shells and web shells. Backdoors enter websites using Trojans. Less commonly, the Trojans themselves act as backdoors. The main purpose of backdoors is to stealthily control your computer. Typically, a backdoor allows you to copy files from an affected computer and vice versa, transfer files and programs to the affected computer. In addition, the backdoor usually allows you to remotely access the registry, perform system operations (reboot the PC, create new network resources, modify passwords, etc.).

The backdoor essentially opens a back door for the attacker to the user's computer. Backdoors are used to steal confidential information from personal computers, such as stealing mail passwords, credit card information, access to payment systems, SSH, FTP, administrative control panels, etc. The threat of Backdoor has increased recently due to the fact that many modern network worms either contain a Backdoor component or install it after infecting a PC.

There are 3 types of shell access:

- "BindShell" - the most common, works on the "client-server" architecture, that is, the backdoor is waiting for a connection.
- "Back Connect" - used to bypass firewalls, the backdoor itself tries to connect to the hacker's computer.
- "Middle Connect" - the backdoor and the hacker's computer exchange data through an additional server [4].

It is more difficult to detect backdoors that involve changing object code rather than source code — object code is much more difficult to verify because it is designed for machine readability, not human readability. These backdoors can either be inserted directly into the object code on disk, or inserted at some point during compilation, assembly linking, or boot-up - in the latter case, the backdoor never appears on disk, but only in memory. Backdoors in object code are difficult to detect by inspecting the object code, but they are easy to detect by simply checking for changes (differences), especially in length or checksum, and in some cases they can be detected or analyzed by disassembling the object code. In addition, object code backdoors can be removed (provided the source code is available) by simply recompiling from the source code.

Thus, for such backdoors to escape detection, all existing copies of the binary must be cracked, and any checksums of the check must also be compromised, and the source must be inaccessible to prevent recompilation. Alternatively, these other tools (length checker, comparison, checksum, disassemblers) can themselves be compromised to hide the backdoor, for example by discovering that the corrupted binary is computed from the checksum and returning the expected value rather than the actual value. To hide these further disruptive activities, the tools must also hide the changes in themselves - for example, a corrupted checksum must also detect if it counts itself (or other disruptive tools) and return false values. This results in extensive changes to the system and the tools needed to hide one change.

Since the object code can be regenerated by recompiling (rebuilding, re-linking) the source code, creating a permanent backdoor of the object code (without changing the source code) requires undermining the compiler itself so that when it detects that it is compiling the attacked program, it inserts the backdoor - or, alternatively, an assembler, linker, or loader. Since this requires undermining the compiler, this in turn can be fixed by recompiling the compiler by removing the backdoor embed code. This protection, in turn, can be compromised by placing the original meta-backdoor in the compiler so that when it detects that it is compiling, it then inserts that meta-backdoor generator along with the original backdoor generator for the original program that was attacked. The original meta-backdoor can then be removed and the compiler recompiled from the original source with a compromised compiler executable: the backdoor has been loaded. This attack dates from Karger & Schell and was popularized in article entitled "Reflections on Trust"; therefore, it is colloquially known as the "Trust" attack. See details on compiler backdoors below. Similar attacks can target lower levels of the system, such as the operating system, and can be inserted during the system boot process; they are also mentioned by Karger & Schell and now exist in the form of boot sector viruses.

A traditional backdoor is a symmetric backdoor: anyone who discovers a backdoor can use it in turn. The concept of an asymmetric backdoor was introduced by Adam Young and Motie Young in Proceedings of Advances in Cryptology: Crypto '96. An asymmetric backdoor can only be exploited by an attacker who injects it, even if the full implementation of the backdoor becomes publicly available (for example, through publication, discovery and disclosure through reverse engineering, etc.). In addition, it is computationally difficult to detect the presence of an asymmetric backdoor in black box requests. This class of attacks is called kleptography; they can be executed in software, hardware (eg smart cards), or a combination of both.

A sophisticated form of a black box backdoor is a compiler backdoor where not only the compiler is undermined (to insert the backdoor into some other program, such as a login program), but is further modified

to detect when it is compiled and then inserted as backdoor embed code (targeting another program) and code that alters self-compilation, similar to how retroviruses infect their host. This can be done by modifying the source code, and the resulting compromised compiler (object code) can compile the original (unmodified) source code and insert itself: the exploit was blocked on download [5].

Once a system has been compromised with a backdoor or Trojan horse such as the Trusting Trust compiler, it is very difficult for a "legitimate" user to regain control of the system - usually a clean system needs to be rebuilt and data (but not executables) transferred over. However, several practical disadvantages of the Trusting Trust scheme have been suggested. For example, a sufficiently motivated user might carefully review the machine code of an unreliable compiler before using it. As mentioned above, there are ways to hide the Trojan horse, such as undermining the disassembler; but there are ways to counter this protection too, for example, write your own disassembler from scratch.

A common method for countering trust attacks is called Diverse Double-Compiling (DDC). This method requires a different compiler and source code for the compiler under test. This source compiled by both compilers gives two different stage 1 compilers, which, however, should have the same behavior. Thus, the same source code compiled by both stage 1 compilers must then lead to two identical stage 2 compilers. Formal proof is given that the latter comparison guarantees that the intended source code and executable of the compiler under test will match under certain assumptions. In practice, such checks are not performed by end users, except in the extreme case of intrusion detection and analysis due to the rarity of such sophisticated attacks and because the software is usually distributed in binary form. Removing backdoors (including compiler backdoors) is usually done by simply restoring a clean system. However, sophisticated checks are of interest to operating system vendors to ensure that they do not propagate a compromised system, and in high-security settings where such attacks are a real problem [6-7].

Conclusion and prospects for further research

Thus, this backdoor poses a serious threat. It is not only engaged in cyber espionage, but can also be used for phishing, since it is able to display windows and notifications with any content. In addition, it can download and install any other malicious applications, as well as execute arbitrary code.

It is easy to guess that backdoors used for unauthorized access to a remote computer allow attackers to obtain all kinds of information. This information includes not only various documents that the user is working with on the infected computer, but also his conversations, messages received by e-mail, etc., etc. In addition, the backdoor allows you to stealthily control your computer, modify passwords, remotely access the system registry or application configuration files, reboot the system, etc. At the same time, the most unpleasant danger of a backdoor lies not even in the fact that it completely opens the system to the will of the attacker - ordinary users are rarely so interesting to those who work with backdoors that they seriously talk about the threat to information. Many modern network worms either contain a backdoor component or install it after infecting a computer. This backdoor is then commonly used by cybercriminals to scan for vulnerabilities and compromise the network through the infected user's computer.

References

1. Addicott J. Cyberterrorism: Legal Policy Issues / Jeffrey F. Addicott // Legal Issues in the Struggle against Terrorism / ed. by John N. Moore, Robert F. Turner. – Durham, Carolina Academic Press, 2010. – P. 592.
2. Butuzov V.M. Countering computer crime: some aspects of international experience (on the example of law enforcement agencies of the United States and Germany) / V.M. Butuzov // Information security: man, society, state. - 2009. - No 1. - P. 30–38.
3. Cyber Cold War Looming for U.S. // USA Today [Electronic resource]. – Access mode: http://www.questia.com/library/1G1-245805413/cyber-cold-war-looming-for-u-s
4. Dubov D. Strategic aspects of cybersecurity of Ukraine / D. Dubov // Strategic priorities. - 2013. - No 4. - P. 119–126.
5. Manzhay O.V. The use of cyberspace in operational and investigative activities / O.V. Manzhay // Law and Security. Scientific journal. - 2009. - No 4. - P. 142–149.
6. Marchenko A.V. Social consequences of cyberterrorist danger in the era of information technology / A.V. Marchenko // Methodology, theory and practice of sociological analysis of modern society: collection. Science. works Kharkiv. nat. V.N. Karazin University. - 2008. - No 1. - P. 355–360.
7. Hildreth SA Cyberterrorism: Materials of the Congress Research Service // Cyberwar: Report of the Congress Research Service RL30735 / SA Hildreth [Electronic resource]. - Access mode: http://www.infousa.ru/information/bt-1028.htm.